Virtual Sensor Web Infrastructure for Collaborative Sensing (VSICS): Architecture, Implementation, and Use Cases

Dr. Prasanta Bose and Ryan Timmons
Lockheed Martin Space Systems Company, Advanced Technology Center
1111 Lockheed Martin Way
Sunnyvale, CA 94089 USA

***Abstract*- A key challenge in weather/science analysis and prediction tasks that involve sensing, monitoring and aggregation of events across time, space and sensing domains is dynamic configuration and robust coordination of distributed information sources driven by semantic decision models and user demands. The information source nodes in such a sensorweb architecture correspond to gateways to physical sensors and sensor networks. The NASA AIST funded VSICS project is developing a middleware infrastructure that enables flexible and robust management (binding of services to requests) and distributed coordination of such gateway resources (called virtual sensors in VSICS) that maximize user utilities. This paper presents initial results on implementing such an architecture and applying to prediction of lightning events (a primary cause of wildfires) and exploitation of such events to semantically mark cloud imageries obtained from continuous satellite based cloud-imaging devices for focusing analysis and prediction. We also discuss our efforts in broadening this use case to involve other risk elements for wildfires, including fuel models, long and short term moisture data, and proximity of human habitation.**

I. Introduction

Geoscience and space science research involving exploitation of in-situ sensor networks and remote observing instruments offers an unprecedented opportunity for coordinated observation and science data processing critical to understanding and timely prediction of climate and (earth and space) weather changes. Current efforts [1,2] in specialized virtual sensors or observatories targeting the atmosphere, ocean, carbon management, space weather and earth-science domains raise the challenge to create a common collaborative sensorweb infrastructure that coordinates sensor resources and data products from individual observatories for effective and robust science processing and timely hazard/weather prediction.

Current sensorweb infrastructures (exemplified by in-situ sensor networks and remote space and air-based sensing platforms) fail to dynamically adapt and provide robust performance in context of above tasks. The limitations arise from inflexibility with respect to exploiting new sensing platforms and new information services. Current sensorweb architectures and deployments assume statical configuration at design and deployment time. They are configured and optimized based on specific assumptions about the operating environment and tasks. Run-time adaptive (re)configuration of such systems via addition of new platforms or via changes in the services provided by the platforms is very difficult and costly. There is no universal format for data collected from such sensors which makes integration of disparate sensor networks and prediction/modeling systems difficult.

This paper discusses the Virtual Sensor Web Infrastructure for Collaborative Science (VSICS) Architecture, design and implementation being developed by the Advanced Technology Center of Lockheed Martin Space Systems to meet the above challenges. We present the VSICS challenges and capabilities defined to meet those challenges, results of domain analysis, and current status on implementation.

II. VSICS Challenges

Some of the key requirements and challenges to design of VSICS architecture are:

A. *Autonomous and scalable run-time configuration of services involved in decision-making and physical sensing to maximize user utility and preferences*.

Current and projected sensorweb applications involve or will involve decision analysts who compete for information services provided by the Sensorweb. In such domains sensorweb performance is evaluated on the basis of the services delivered to the decision analysts and Utility functions measure the effectiveness of the delivered service (data collection and processing. The utilities span functional (e.g. wind-vector) and non-functional attributes (accuracy). The services are managed by distributed service providers. The analysts exploit workflow models for their science and hazard prediction tasks. The key problem VSICS addresses is decentralized management and scheduling of the resources or services that lead to maximization of user objectives specified by workflow-based model of the task.

B. *Adaptive coordination of resources and services for robust performance.*

Given the dynamics of sensor systems with chance of network link or sensing node failure, and the dynamics of the task domain (e.g. rapid spreading of forest fires), a key problem is adaptive closed-loop coordination to assure robust and effective performance of the system. Addressing such a problem requires ability to perform joint networking and sensing resource management decisions, intelligent monitoring for closed loop control and coordination at all levels of the system decomposition.

C. Dynamic composition for flexible plug-and-play.

The scalability of a sensorweb depends on its flexibility to compose services. Low-level approaches to composition fail to provide the abstractions needed for utility-driven run-time composition. A key problem is development of information-oriented abstraction of services at all layers that enable flexible and efficient composition of services (both information processing and physical data sampling).

D. Ease of extensibility to new services.

A key goal is middleware easy enough to use and understand such that researchers in earth science disciplines can not just search for services and run workflows, but create their own. We intend to provide easy to use tools to allow scientists to quickly connect their own sensor networks or modeling software to the VSICS network, making them available for collaborative efforts.

III. VSICS Architecture and Capabilities

VSICS is composed of two complementary layers of service. One is the Physical Sensor Layer (PSL) which deals with the behavior of the sensor nodes and the formation of networks between sensor nodes to jointly collect data. The other is the Decision Support Layer (DSL) which consists of higher-level services. These services include databases, prediction services, and other applications that use the data from the sensors and also decide how the sensors should be used. Following the loss of the second year of funding focus was shifted away from the PSL and this paper will deal mainly with the DSL.

To enable a robust, decentralized framework we structure our architecture around *service registry&Coordination agents.* Each *registry&Coordination agent* is responsible for managing one or more VSICS services and provides the frontend interface used to retrieve information from the service. These agents are networked together allowing for distributed execution of workflows among services in different locations.

We extend and exploit the Ptolemy framework for actor-oriented and data-driven programming [3] of the agents and their workflow-driven coordination. The distributed integration of the Ptolemy-based actors is developed by building upon the JXTA set of peer to peer networking protocols [4]. To illustrate VSICS conops, we present a walkthrough of a cycle of execution in VSICS.

A. Registry initialization and discovery

Each registry is started and uses the previously mentioned features of JXTA to automatically discover other registries and form communication links to them.

B. Workflow Search

A user specifies a workflow to be completed by the search process. This is done by selecting one or more services from those available.

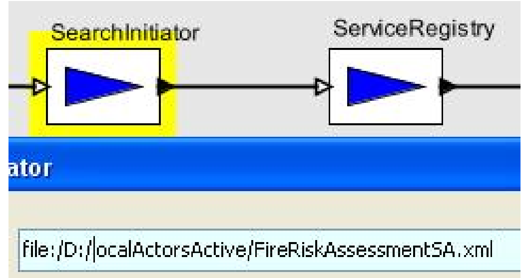

Fig. 1. Selection of a service to begin a workflow search.

We expect that most use will involve a user picking a single service with the desired end product as an input as shown in Fig. 1.

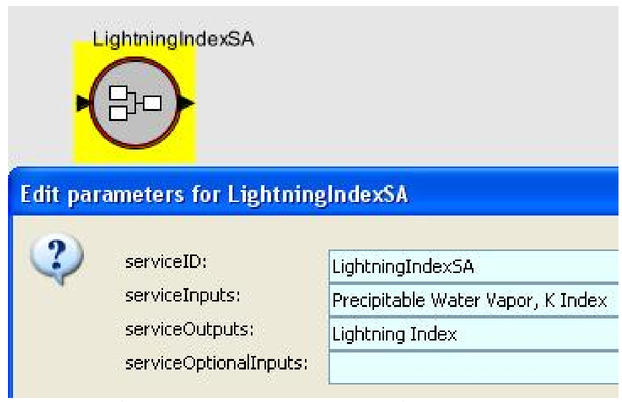

Fig. 2. View of the input and output parameters of an example VSICS service.

Services can have required inputs, optional inputs, and outputs as shown in Fig. 2. The search process attempts to find the largest complete workflow; a workflow is considered complete when all member services have providers for all their required inputs. The algorithm recursively adds any services on the local registry that match an output to an input in the existing set of services or match an input to an output and is repeated across all connected registries.

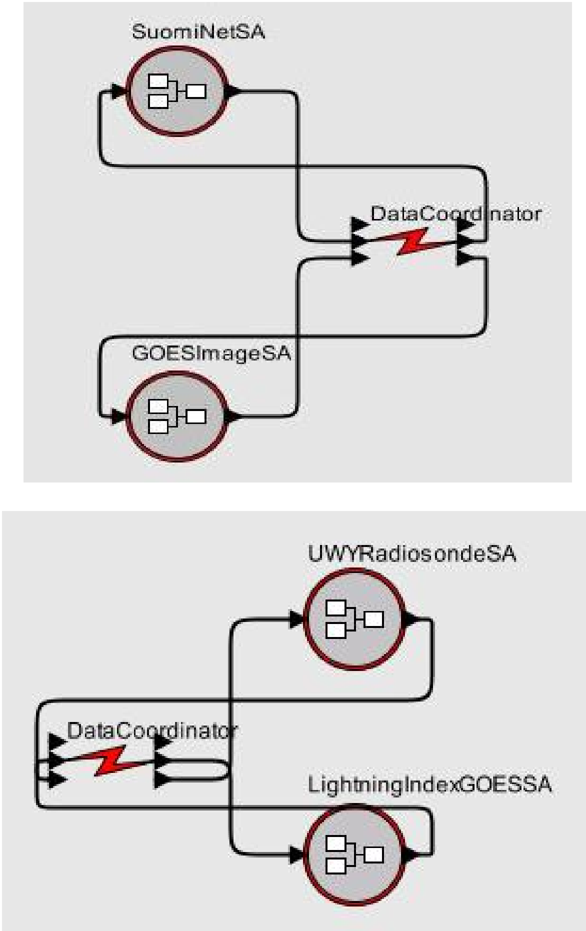

Fig. 3. Views of two halves of a distribute workflow, each running on a separate computer.

Once the search is completed, messages are sent telling each registry to begin execution of a partial workflow consisting of whatever services in the whole workflow that are hosted on that particular registry; instructions on what registries host the other services are also included. Examples of a workflow distributed across two registries are shown in Fig. 3.

C. Preparation for Distributed Execution

Each partial workflow has a DataCoordinator actor which manages its communication. For messages between services on the same registry, the coordinator essentially acts as a switch. The coordinator has basic routing capabilities that it uses for sending messages to services on other registries.

D. Execution

Execution messages in our data-driven model fall into two general categories – requests and data. Data travels forward in the publish/subscribe model (a service providing datatype X sends to services requiring X) while requests go the other way. It is expected that a data message will be provided in response to a request, but data can also be sent without a request prompting. The content of these messages can also be metadata. We provide a standard metadata format that includes a range of times and a set of locations for which a service is available. This format will eventually hold information on nonfunctional attributes as well.

Although they are not required in order for communication in VSICS, we provide two templates for communication. One is a data "pull task", which consists of a set number of data requests. The service running the pull task sends out all queries and waits until all responses are received, then does some programmer-specified processing. A simple example would be a request for data over some time range which would then be processed statistically when it has all been received. The second model is a data "push task", where service that provides some type of data is set up to periodically send the latest version. In this fashion a service can provide the latest data or predictions as they become available. Push tasks are generally initiated in response to a specific request for continuously updated data.

We assume that the user's provided actor to begin a search will contain whatever code is necessary to output and/or display results of the workflow to the user.

IV. VSICS USE CASES

We have applied VSICS to realize two related use cases. These use cases emphasized publicly available but disparate data sources and creation of VSICS service to provide a common interface to integrate them coupled with established scientific models to allow useful processing and synthesis of the data sources.

A. Lightning Prediction

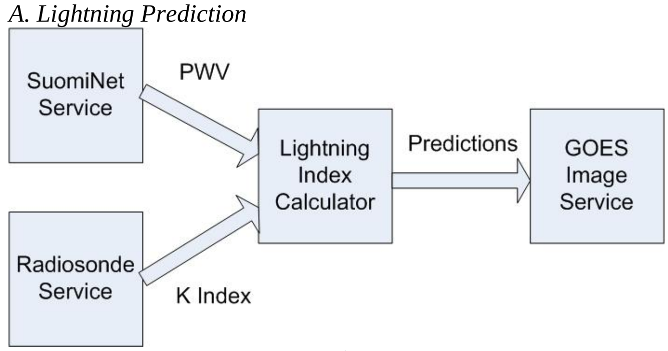

Fig. 4. Conceptual view of the lightning use case.

One use case involves lightning prediction, with a data flow depicted in Fig. 4 above. We implement a previously developed logistic regression model using four types of weather data – precipitable water vapor (PWV), 9-hour change in PWV, K-index (derived from drybulb and dewpoint temperatures at various pressures above sea level) and atmospheric electric field [CITE]. Their model predicts the onset of lightning activity within 90 minutes to 12 hours when the index drops below a certain threshold. We have designed services to implement a simplified version of this model at 39 sites across North America. We could not find a suitable data source for atmospheric electric field but were able to use the SuomiNet project and the NOAA's GPS-MET stations to access PWV data [5,6]. We use a collection of radiosonde sites maintained by the University of Wyoming to access the K index[7]. The observation stations for these services are in general not the same but we found 52 sites with stations close enough to make predictions. For each data source, a service was written that maintains databases of the relevant data and automatically downloads data updates as they become available. Another self-authored service combines the input data to compute the index.

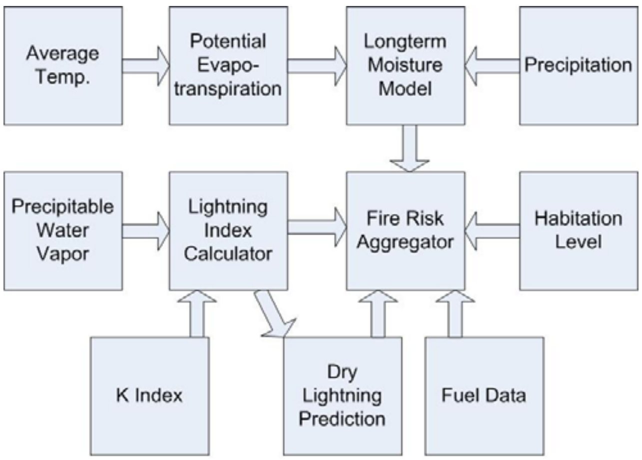

Fig. 5. Conceptual view of the wildfire use case

B. Fire Hazard Prediction and Warning

The second use case shown above builds upon the first. We seek to combine information about multiple risk factors associated with wildfires, in particular lightning-ignited fires. Our lightning prediction index from the previous use case is one input, and is the most frequently updated. Also included is a consideration of the probability of lightning without significant precipitation which is much more likely to start fires. We hope to use data from the US Forest Service's Smoke and Fire group [8] in our use case but are currently waiting for them to finish a system upgrade; in the meantime we use an approximation of their algorithm for discriminating dry and wet convective days based upon a certain dewpoint depression and temperature difference

Combined with this data on lightning events are three long-term factors involved in fire risk. One is a model of the moisture surplus/deficit, for which we use C.W. Thornthwaites's model of potential evapotranspiration [9] combined with precipitation data from the National Weather Service's Climate Prediction Center [10]. A second is a measure of the average fuel level in an area. ISO's FireLine service provides fuel measurements across nine western US states [11]; unfortunately time and budget constraints prevented us from fully integrating their data. Finally, to assess relative hazards from fires we use our own model of potential impact on human habitation. Using data from the 2000 census, for each location a score is computed based on weighting of population centers by size and proximity to the fire. This method is very simple and does not take into account any models of fire spread.

Unlike the lightning prediction use case, we do not have an established model of how the various risk factors described above interact, and could not provide definite predictions such as probability of a wildfire or expected number of fire events even if we had accurate data for all of the sources enumerated. Thus for the moment our central service aggregates and displays the data from the input services but does not process the data like the central actor in the lightning prediction use case did.

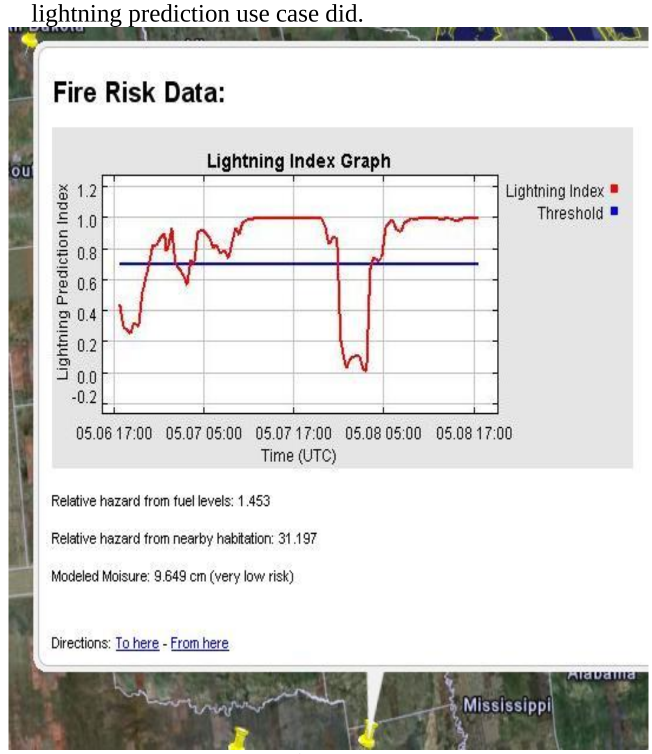

Fig. 6. Collected data for one location near Shreveport, La.

Currently this actor combines the data and displays it in Google Earth for each location as shown in Fig. 6.

V. SUMMARY AND FUTURE DIRECTIONS

We have implemented decentralized search for workflow configuration and decentralized execution of the workflow. Many services have been created from existing sensor and data sources and they have been combined into use cases for lightning prediction and wildfire prediction/risk assessment. Future VSICS work will focus on:

A. Interface and Use

Currently our demonstration version requires the user to run service registry on their local machine. If we continue to a finished version users will be able to run a thin client that connects remotely to a service registry, and some machines will be dedicated servers. The client will get the services responsible for user input and output and run them, but the bulk of execution will take place on the remote registry machines. A development goal would be a client able to run on some PDA (e.g. smart phones) so that responders to severe weather events could use VSICS services in the field. Users will also be able to use VSICS from anywhere on the internet (current version was built for use only within Lockheed's network). Users will also have more choices in selecting a workflow to run; the current system selects the largest workflow possible. In the future the user will have choices as whether to use optional/redundant services and nonfunctional attribute information to help make those sorts of choices.

B. Improvement of Service Authoring

We developed template code that could be quickly modified to make simple services and had templates of the appropriate Ptolemy actors. While these tools work well enough for the developer, allowing the creation of a service actor from a service backend implementation in as little as 10 minutes, they still require some knowledge of Java and Ptolemy to use. We hope to provide better tools in a future version as well as better support for creating links to non-Java service implementations.

C. Reconfiguration:

We would like the ability to have users manually add new services into a workflow during execution, and have the coordinators determine the new links to be formed based on the added services' publish/subscribe attributes. Another goal is automatic load balancing between multiple users. VSICS should be able to monitor the demand when multiple workflows are using the same service, and provide for alternate services and/or prioritize workflow access if a service is being overloaded with requests. This could apply to sensor services with finite power and other resource limits, or to prediction/simulation services that require time and computing resources to handle each request. The load balancing will hopefully be handled as one of the nonfunctional attributes (see below).

D. Additional Data Sources:

As was mentioned in the detailed explanation of the wildfire use case, time and money constraints prevented us from integrating the ISO FireLine fuel data [11]. This product appears to be the best fuel model available over a broad geographical area (9 states in the western U.S.) and would allow for accurate data in that region. Dr. Miriam's Rorig's work on classification of dry convective days will also be useful once their service is restored again [8]. Finally, we would want to obtain data from the World Wide Lightning Location Network [12]. This would provide a confirmed source of lightning strikes into the fire risk workflow as well as allowing us to train the lightning predictor, perhaps even to the point of having location-dependent weightings of the predictors.

Acknowledgment

We gratefully acknowledge the funding of NASA's AIST program for the development of VSICS.

References

[1] G. Eason, B. Noble, and I.N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," *Phil. Trans. Roy. Soc. London,* vol. A247, pp. 529-551, April 1955.

[2] J. Clerk Maxwell, *A Treatise on Electricity and Magnetism,* 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68-73.

[3] I.S. Jacobs and C.P. Bean, "Fine particles, thin films and exchange anisotropy," in *Magnetism,* vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," *J. Name Stand. Abbrev.*, in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," *IEEE Transl. J. Magn. Japan*, vol. 2, pp. 740-741, August 1987 [*Digests 9th Annual Conf. Magnetics Japan,* p. 301, 1982].

[7] M. Young, *The Technical Writer's Handbook.* Mill Valley, CA: University Science, 1989

[1] D. De Roure, N. Jennings, N. Shadbolt, and M. Baker, "Research agenda for the semantic grid: A future e-science infrastructure," Technical Paper, 19 February 2002, http://www.semanticgrid.org/html/semgrid.html.

[2] B. Ludäscher, Arcot Rajasekar, Joseph Goguen, David Stockwell and Victor Vianu, NSF ITR (large): "Enabling the science environment for ecological knowledge (SEEK)," October, 2002, https://www.fastlane.nsf.gov/servlet/showaward?award=0225674.

[3] Edward A. Lee, "Overview of the Ptolemy Project," *Technical Memorandum No. UCB/ERL M03/25*, University of California, Berkeley, CA, 94720, USA, July 2, 2003.

[4] Sing Li, "JXTA 2: A high-performance, massively scalable P2P network, " Wrox Press, Nov. 2003, http://www.ibm.com/developerworks/java/library/j-jxta2/. See also https://jxta.dev.java.net/.

[5] Ware, R.H., et. al., "SuomiNet: A real–time national GPS network for atmospheric research and education," *Bull. Amer. Meteor. Soc.*, 81, 677–694. See also http://www.suominet.ucar.edu/.

[6] Ware, R., M. Exner, et al., "GPS sounding of the atmosphere from low Earth orbit - preliminary results," in Bull. Am. Meteorol. Soc., 77, 19-40, 1996. See also http://www.cosmic.ucar.edu/gpsmet/.

[7] See http://weather.uwyo.edu/upperair/sounding.html.

[8] Rorig, M.L., S.J. McKay, S.A. Ferguson, and P. Werth, "Model-generated predictions of ddry thunderstorm potential," in *J. Appl. Meteor. Climatol.*, 46, 2007, 605–614. See also http://www.fs.fed.us/pnw/airfire/sf/.

[9] C.W. Thornthwaite, "An approach towards a rational classification of climates," in Geog. Rev. 38, 1948; 55-94.

[10] See http://www.cpc.ncep.noaa.gov/products/analysis_monitoring/cdus/prcp_temp_tables/.

[11] See http://www.iso.com/index.php?option=com_content&task=view&id=839&Itemid=524

[12] See http://wwlln.net/.